Summer 2021

Adapting Open Pit Mine Design Fundamentals to Leverage the Advantages of Autonomous Haulage Systems

Race Owens

Adapting Open Pit Mine Design Fundamentals to Leverage the Advantages of Autonomous Haulage Systems

A publishable paper submitted in partial fulfillment
of the requirements for the degree of

Master of Science, Mining Engineering

By
Race B. Owens
August 2020

Montana Technological University
Mining Engineering Department

Abstract

It is common practice, and even legally required in many jurisdictions, to design two-lane haul roads in open pit mines to three and a half times the widest dimension of the haul trucks operating on the road. In open pit gold mines with high strip ratios, the road width has a significant impact on the economics of a design. It is possible to minimize the flattening of the highwall if the road width is reduced, assuming the width used is not needed to flatten the slopes for geotechnical purposes. With the use of autonomous haulage and pull-outs, it may be possible to operate safely and efficiently on a reduced road width of two times the width of the hauling equipment.

Introduction

Autonomous haulage fleets have been successfully implemented in large copper, iron ore, and oil sands mines with low strip ratios; however, the demand has not been as strong in high strip ratio mines. This paper proposes that the adoption of autonomous systems may have additional economic benefits associated with mine design changes that have not yet been accounted for in the analysis of these systems. Strip ratio is a ratio that expresses how much waste is mined per unit of ore. In mines with low strip ratios, maintaining a full road width for two-lane traffic is simple and has less of an effect on the economics of a design. In mines with less disseminated deposits and significantly higher strip ratios, the road width has a more significant impact on the economics of a design. By reducing the operating road width, the overall highwall angle may be able to be increased and ultimately reduce the strip ratio. This reduction in width may be made possible with haulage automation, assuming the road width used is not needed to flatten the slopes for geotechnical purposes.

The mining industry's standard practice is to design two-lane haulage roads as three and a half times the widest dimension of the haul trucks operating on the road. With autonomous haulage, it may be possible to reduce the road width to two, or less, times the truck width with pull-outs for two-way traffic. In this paper, the requirements from an autonomous fleet to operate on reduced width of in-pit ramps are examined. With the use of autonomous haulage, operations could occur on these roads safely and efficiently.

Background

Automated Haulage

Parreira (2013) recognized that "In mining, automation is playing an increasingly important role due to a scarcity of high demand metals and skilled personnel to operate the processes. Challenging locations and harsh environments are becoming normal for new ore bodies and mines, so automated systems may become essential." These difficulties can be accentuated for mines located in remote areas. Currently, where favorable conditions in weather and infrastructure are present, automation is thriving. As automated mining technology progresses, the use of automation is becoming more prevalent in all environments. These technologies are becoming common in the mining industry because of the increases in safety and operating efficiencies they bring to operations.

Mines are currently automating processes and equipment to capture the benefits of eliminating the "human factor." When discussing automating machines, it is essential to specify the level of automation being used to ensure clarity. Voronov et al. (2020) presents and describes the following four levels of automation of mining equipment:

1. Remote control
2. Telemetric control
3. Partial automation
4. Full automation

Remote control is used to control machines where manual operation is dangerous. The machine is in the field of view of the operator who uses the remote-control device. Telemetric control is similar to remote control, but the operator is located at a considerably farther distance, not allowing line of sight operation. The use of cameras and sensors enables the operator to control equipment. Partial automation is where one operator regulates the activities of several units of equipment. Certain aspects of the equipment's operation move according to an algorithm, and other factors still need to be controlled by an operator. Finally, equipment works completely independently without human participation when it is fully automated, relying on preprogrammed algorithms and artificial intelligence to control the machines. This independent operation shifts the operator's job from operating the equipment to managing the equipment's operation and monitoring efficiency. Human intervention with fully autonomous operations is only required when serious problems arise.

The autonomous haul trucks referenced in this paper are considered to be at the highest level, fully autonomous; however, other mining equipment is commonly implemented at the other levels. Inputs such as dispatch, GPS(Global Positioning System), LiDAR(Light Detection And Ranging), and proximity sensors provide data that can result in highly precise, independent operation. The consistency and precisions provided by these systems allow these trucks to operate at a much higher level of safety than equivalent human-operated haul trucks. Removing manually operated haul trucks also increases the consistency of operations and removes humans from potentially hazardous areas and situations. According to Parreira (2013), at mines where autonomous haulage has been implemented, accidents due to human error and poor driving habits have been eliminated.

In the last few years, autonomous mining equipment has been an unavoidable topic when mentioning technological advancements in the mining industry. Even though automated mining is still at the forefront of mining technology, the first attempts to automate mining equipment were made in the 1960's but it wasn't commercially viable until the 2000's. The following timeline summarizes the steps Komatsu and Caterpillar, the main manufacturers of haul trucks investing in unmanned mining trucks, took to implement this technology commercial.

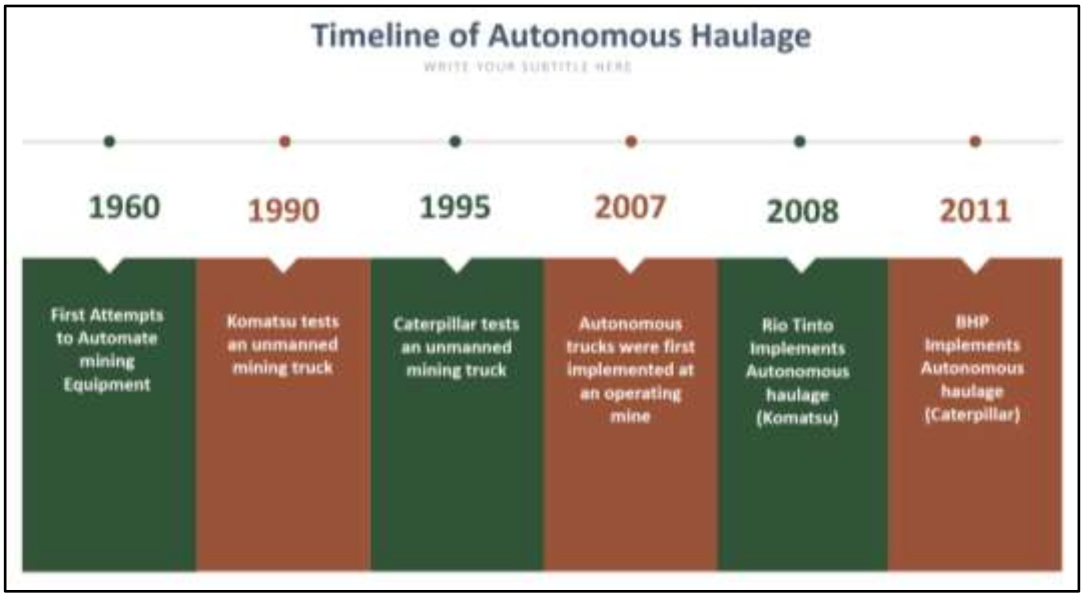

Figure 1: Timeline for Autonomous Haulage 7 (Voronov, Voronov, & Makhambayev, 2020)

Both Caterpillar and Komatsu have their own autonomous haulage systems. Caterpillar's autonomous haulage system is named Command for Hauling. The trucks perform more independently within this system, with each truck routing and tracking possible interactions with other vehicles. The centralized control center is mainly responsible for the allocation of trucks between loading and dumping points. Command for Haulage is more decentralized, resulting in the onboard equipment of these unmanned trucks being more complicated. Caterpillar automated trucks are equipped with both LiDARs and RADARs(Radio Detection And Ranging) to help their trucks make these complicated decisions. Komatsu's haulage system is named FrontRunner. Komatsu's approach to an automated haulage system is to have a more centralized design with more of the decisions made by the control center. Haulage trucks are monitored individually in an unmanned zone, with the system paving each truck's route as it assigns trucks to loading and dumping points. While making these decisions, the system also monitors any possible interactions between unmanned and manned machines. With fewer decisions made onboard, each truck is only equipped with a RADAR and conventional rangefinders to locate the road's edge. Although these systems are vastly different in their approach, they both fall into the highest level of automation, fully autonomous. (Voronov, Voronov, & Makhambayev, 2020)

As more mines turn to autonomous haulage, the benefits such as reduced operating costs, increased precision, and increased safety meet or exceed expectations. Compared to manually operated trucks, autonomous haulage reduces fuel consumption, eliminates expensive operating labor, decreases maintenance costs, increases the life of the equipment, reduces capital costs, and reduces harmful emissions into the atmosphere (Voronov, Voronov, & Makhambayev, 2020). Along with reducing costs, safety is improved since the vehicles are operated consistently and precisely with fewer humans placed in potentially hazardous operating areas.

Current Design Fundamentals

Current operating widths for haulage roads are based on the principle that there needs to be half a vehicle width on either side of each traffic lane for improved safety. Another aspect that adds to the total width of a haul road is the mandatory berm, which the Electronic Code of Federal Regulations(e-CFR) (2021) states is required on the banks of roadways where a drop-off exists of sufficient grade or depth. These mandatory "berms or guardrails shall be at least mid-axle height of the largest self-propelled mobile equipment which usually travels the roadway." (e-CFR, 2021). Additionally, drainage ditches and bench accesses are commonly added to a haulage ramp, contributing to the total road width.

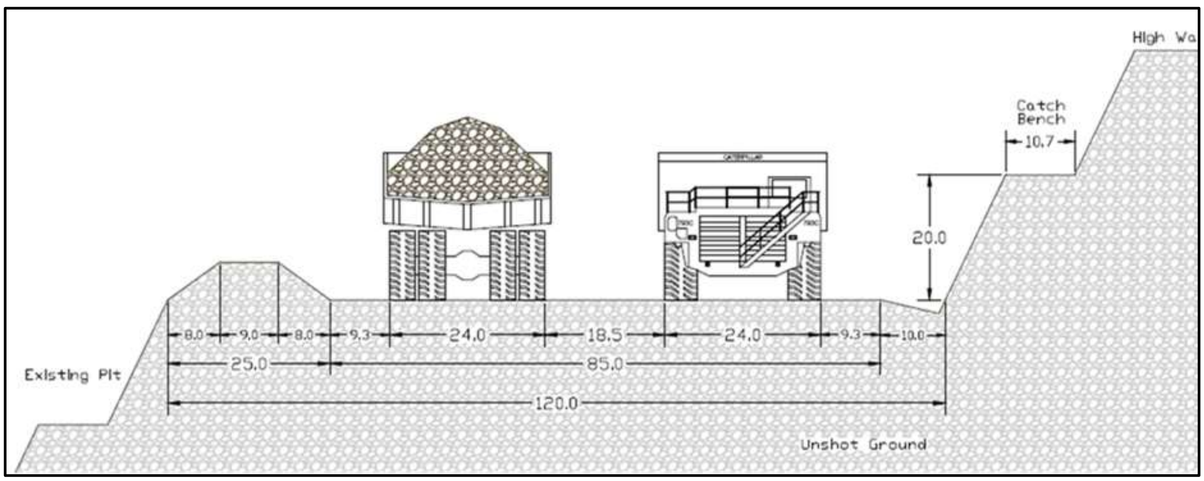

Figure 2: Example Haul Road Cross-section (Masabattula , 2011)

After designing a cross-section and determining the total width of a road, factors such as the positioning of the road, the number and location of switchbacks, and the intensity of the gradient will also directly affect the economics of the mine design. An example haul road cross-section can be found in Figure 2. A haul road gradient of 10% is commonly used for haulage roads in open pit metal mines. Mine engineers must choose a road gradient that balances increased operating costs and decreased haulage distances resulting from steeper haulage gradients. Also, laws and regulations may dictate a maximum haulage ramp gradient for safety reasons (Collins, Fytas, & Singhal , 1987). Generally, the road gradient is based on the haul trucks' capability and what the manufacturer suggests for optimal equipment operation.

Safety is an immense concern when creating a design. Many aspects of haul road design are dependent on managing or maximizing safety while still operating efficiently and productively. This balance of safety and productivity is especially a concern when human-operated vehicles are on the ramps. The vehicle operator should be able to see ahead of the vehicle at the same distance as the maximum stopping distance of the vehicle. (Collins, Fytas, & Singhal , 1987) This fundamental gives the machine enough time to stop to avoid collision with other objects, such as humans, other vehicles, and large rocks. Even with fully autonomous vehicles, the onboard sensors need to "see" far enough to stop before encountering unexpected objects. Operator safety is the most significant concern, along with minimizing vehicle damage. For these reasons, the e-CFR (2021) states, "Water, debris, or spilled material on roadways which creates hazards to the operation of mobile equipment shall be removed." Mining operations commonly use graders to remove obstacles, such as loose rock, snow, and uneven surfaces from haul roads, increasing safety and decreasing rolling resistance. (Collins, Fytas, & Singhal , 1987)

Road maintenance is essential to a mining operation to ensure the safety of those that travel on it, decrease the maintenance cost, and increase the operating speeds. Autonomous haulage operations require consistent, quality road maintenance. Unmanned haul trucks struggle to quickly respond to an uneven road and loose rock on a haul road, resulting in increased fuel consumption and tire wear. (Zhao & Bi, 2020) For these reasons, minimum road width must allow for enough space for haul trucks to operate unaffected by operations such as grading. Like water trucks for dust control, certain operations may not affect the minimum road width since they operate on the same route and at similar speeds as haul trucks.

Previous Work

The author has found no previous work directly relating to changing surface mine design parameters to leverage the benefits of autonomous equipment, specifically autonomous haul trucks. However, there is similar work relating to single-lane traffic in underground mines, train scheduling, and the effects of road parameters on open pit designs.

Underground Haulage

The work pertaining to single-lane traffic in underground mines has its own set of challenges with the limited connection between and directly to the machines. Even though there is better connectivity in surface mines, some of the concepts from operating two-way traffic on a single-lane road with pull-outs in an underground mine are useful. In the papers from Anderson (2019) and Åhlén (2014), the goal was to minimize the time used in reversing to prebuilt pull-outs. Andersson (2019) evaluated using an algorithm to determine which pull-out to use, maximizing the distance traveled while minimizing the delay, time waiting in the pull-out. Many of the underground models involve having one vehicle stopping at the pull-outs and waiting for the vehicle with the higher priority to pass by unaffected by the interaction between the vehicles. This interaction is not optimal but is preferred to collisions or stopping and reversing to the nearest pull-out.

Train Scheduling

While the concepts of pull-outs are relatively common, train scheduling on a single track is more closely related to the idea presented in this paper. Similar to a single track, surface haul roads are costly and difficult to move once built. Like a railroad, the road is created to minimize the capital cost required to build it while maximizing the traffic and productivity on the road. The high investment and complex physical geography of railway construction may be limited by capital budgets, resulting in the frequent use of two-way traffic on single-track railways. In recent decades, the railway industry has focused significant effort on researching train scheduling methodologies to make the single-track railways work efficiently (Xu, Li, Yang, & Gao, 2019). Even though the purpose of this paper was to propose an efficient heuristic approach for the train scheduling problem, it provides useful insights that relate to two-lane traffic on a single-lane road. Xu et al. (2019) addresses train scheduling and some of its intricacies. Note that track pull-outs provide the possibility for two trains to travel parallel to each other while passing, which allows for flow from both directions at the same time. Xu et al. assumed that the positions of individual vehicles are known. Another similarity between train scheduling and haul truck scheduling is the location for both is always known. Even with all these similarities, efficiently operating haul trucks on reduced haul road widths is different from train scheduling. When scheduling trains, some of the pull-outs are used to store parked trains, whereas, in mining, the haul trucks are always traveling when they are on the haul road for productivity reasons. Another difference between the mining industry and the railroad industry is how priority is assigned to each vehicle.

Road Parameters

Sakantsev et al. (2018) addresses minimizing the effect of a haul road on the overall high wall angle by increasing the pit haul road gradient. Issues related to adjusting haul road gradients such as increased maintenance cost and the effect of weather are discussed. This paper only mentions the impact that in-pit haul roads have on the high wall angle. However, the discussion related to road gradient is helpful since reducing haul road gradient can have similar beneficial effects on the overall high wall angle as decreasing road width.

Alegre et al. (2019) also addresses the effects of haul road parameters on the strip ratio of an open pit mine. In this paper, the authors address the impact of three major parameters of haul road design; road widths, switchbacks, and road gradients on strip ratio and cost in both high and low strip ratio pits. Even though increasing the gradient of the road has the largest impact out of the three variables looked at, it also results in a higher maintenance cost. Sakantsev et al. (2018) also found that steeper gradients of access roads will ensure reduced stripping cost but also raise the cost of the transportation.

Along with increased maintenance costs, increasing the haul road gradient may be difficult with current rules and regulations. As Collins et al. (1987) states, mine operators must choose a road gradient that balances increased operating costs and decreased haulage distances resulting from steeper haulage gradients. Laws, regulations, and equipment warranties may also dictate a maximum grade of haulage ramps for safety reasons.

Case Study Methodology

Pit Design

Multiple pits were modeled on data provided for a gold deposit in Montana to demonstrate the effects of haul road width on high strip ratio open pit mine designs. Pit limit analysis has been previously conducted for this deposit using the software program Whittle. The pit limit analysis for this deposit would be mined multiple phases, Figure 9 in the Appendix, but only one phase was analyzed in this case study. Pit designs for this phase were generated for the following haul road widths, zero feet (i.e., no haul road), 65 feet for a traditional single-lane haul road, and 120 feet for a conventional double-lane haul road. These designs can be found in Figure 3. The haul road widths are based on the assumption that CAT 793 trucks will be operated at this mine.

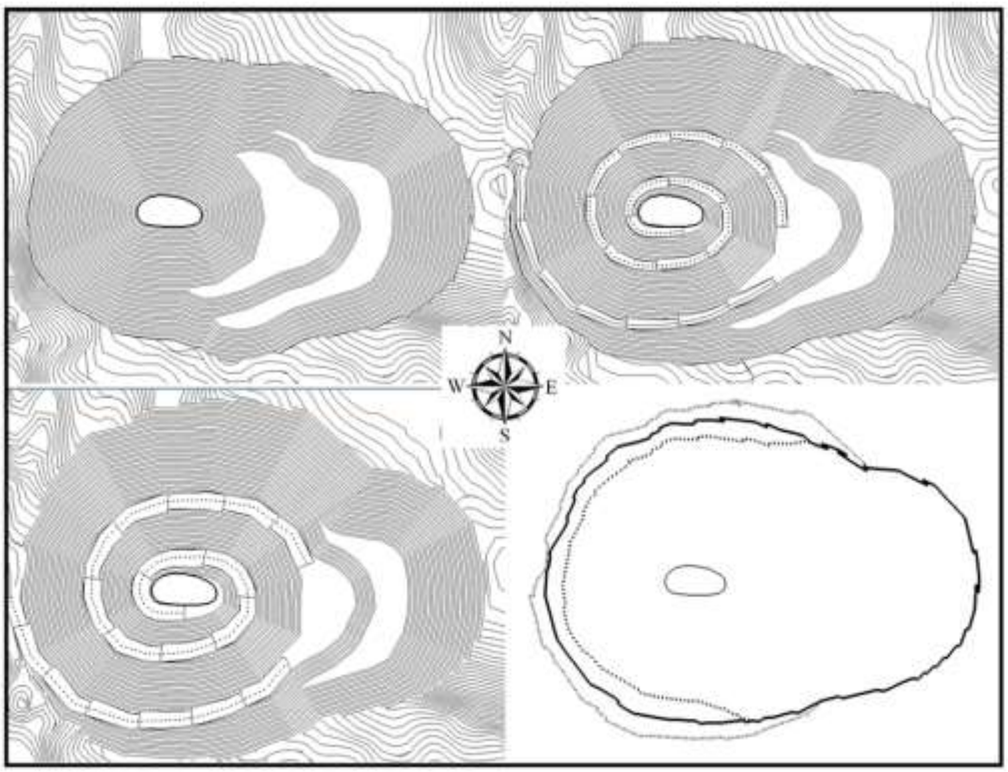

Figure 3: Case Study Pit Designs (Upper left: no haul road, upper right: single-lane haul road, bottom left: double-lane haul road, bottom right: top and bottom polygons of all three designs)

The pit was split into multiple zones that had different geotechnical design restrictions. These Geotechnical parameters were coded into the block model to create a design that falls within the parameters. The design parameters can be found in Tables 1 and 2.

Table 1: Case study pit geotechnical parameters

Geotechnical Parameter	Zone 1	Zone 2				Units
		North East	South East	South West	North West	
Inter Ramp Angle	43	42	44	50	45	[°]
Berm Height	40	40	40	80	40	[ft.]
Berm Width	24	25	22.8	30	17	[ft.]
Batter Angle	65	65	65	65	60	[°]

Table 2: Case study pit design parameters

Design Parameter	Value	Unit
Truck	793	None
Truck Width	27	[ft.]
Double Lane Width	120	[ft.]
Single Lane Width	65	[ft.]

Cross-sections were examined at 22.5-degree angles centered on the lowest bench in the design to work as a visual representation of how the haul road dimensions affect the overall highwall angle and overall pit design. As seen in Figure 4, cross-sections, and additional cross-sections in the Appendix, highwall flattening occurs when larger road widths are implemented.

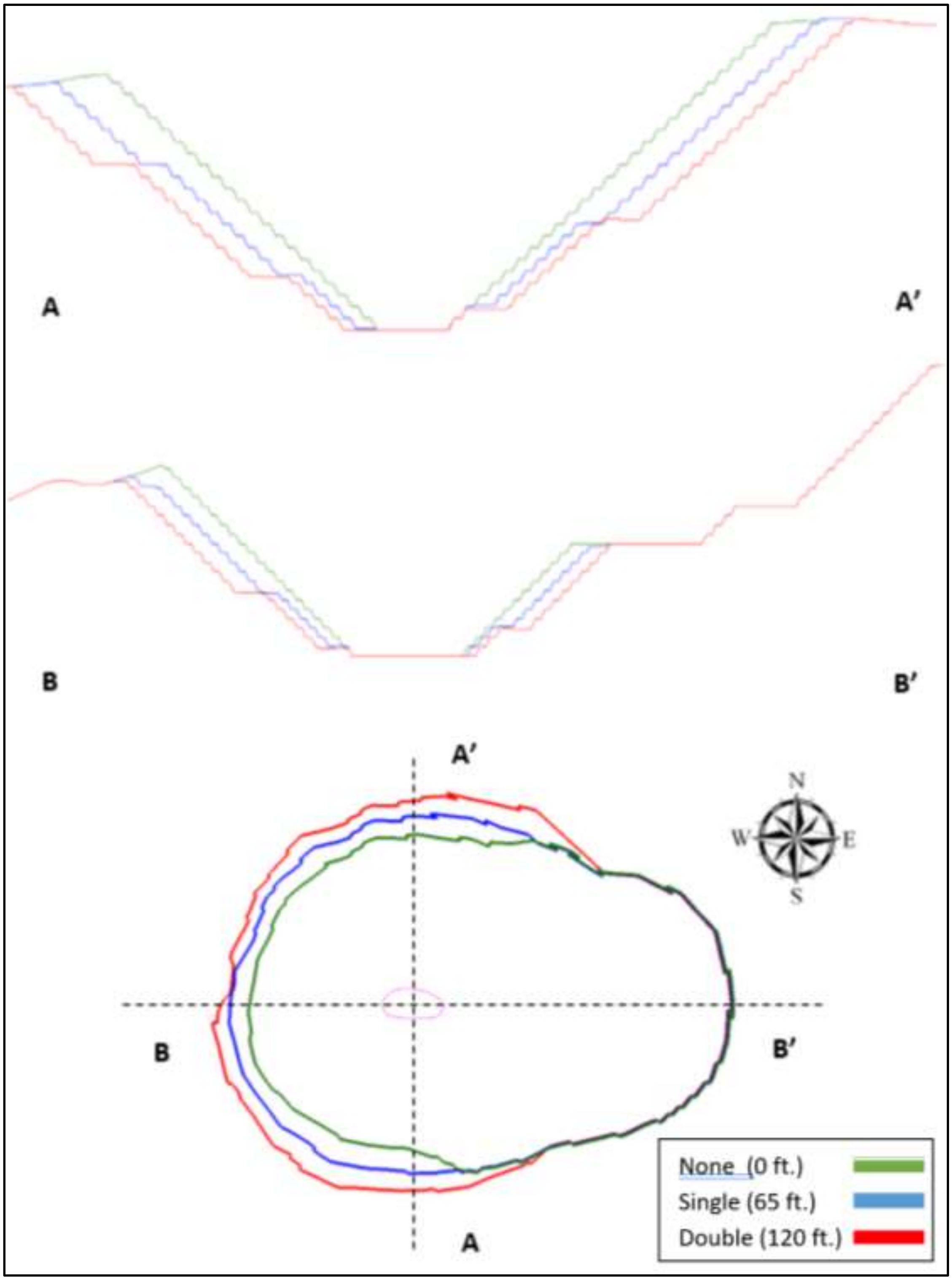

Figure 4: Cross-sections and Cross-sections on the Pit Plan View

The following design considerations were taken to ensure the results were only influenced by the road width parameter:

- The phase was designed in an isolated manner, and the interaction between pits was not considered;
- Pull-outs were not included since, over the entire road, they would result in little additional width;
- Switchbacks were not used;
- Each of the ramps began at the same location on the same starting polygon at the bottom of the pit;
- All seed strings were kept the same to create similar pits. Seed strings determined the shape of geologically constrained areas and where widening is needed to follow the optimal pit shell more closely.

The resulting pit solids from the designs were compared against the block model for the deposit to determine the tons of ore and tons of waste. The strip ratios for these pit shells were calculated and analyzed to determine the effect of haul road width on strip ratio for a final design on this deposit. The pit without a haul road was closest to the Pit shell created in Whittle, and had the lowest strip ratio of 0.90. The single-lane road had a strip ratio of 1.09. As seen in the pit cross-sections, the double-lane road took the most waste and was the largest design while also having the largest strip ratio of 1.32. The double-lane haul road adds 29.8 million tons of waste while only adding an extra 3.8 million tons of ore. Figure 5 shows the total tons mined split by ore, waste, and pit design. This figure also clearly shows the Strip Ratio for each pit. At a mining cost of $2.5 per ton, a double-lane haul road results in an additional mining cost of 84 million dollars more than a single-lane haul road. Wider roads tend to increase the strip ratio in a deposit that already has a high strip ratio, thus increasing the costs considerably.

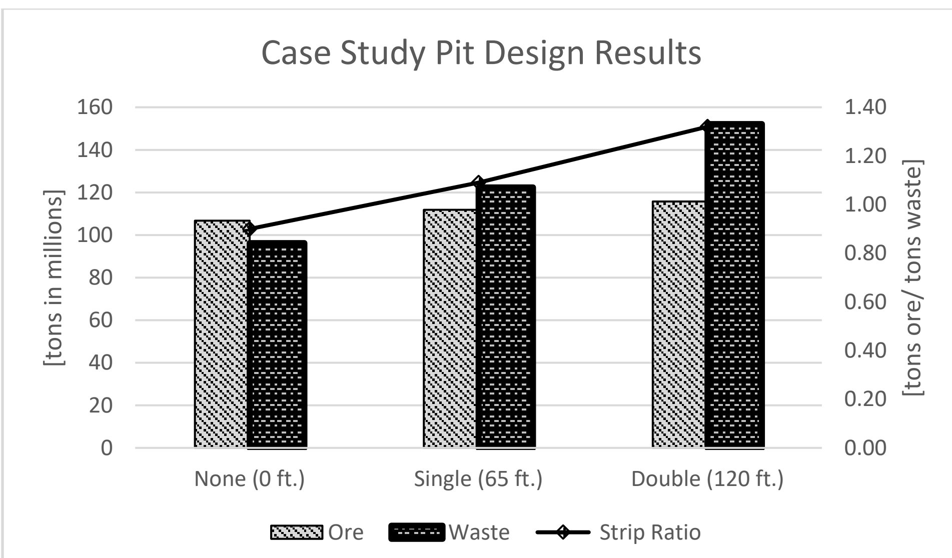

Figure 5: Case Study Pit Design Results

Deterministic Simulation

To model interactions between trucks in the haulage cycle when using single lane haulage, with and without pullouts for the unloaded trucks in the haulage cycle, a deterministic simulation was created using Python 3.8.8 and is included in the Appendix. The default inputs to this simulation are shown in Table 3, Deterministic simulation input variables and default values. The process is shown visually in the Appendix as Figure 18, Flow Chart for Python Simulation Code. In this simulation, the speed, position, direction, and time-delayed for each of the haul trucks are calculated for each second that the simulation was run. The speed of each truck is determined based on the location, speed, and delay values of other trucks, along with the location of the passing areas. The speed of each unloaded truck is calculated so trucks optimally reach pull-outs to incur minimal delay while reducing stoppages, whereas loaded trucks always travel at the maximum possible speed to maximize production. A flow chart showing the process of the optimal speed calculation is shown in the Appendix as Figure 19, Flow Chart for Optimal Speed Calculation Code in the Simulation.

Table 3: Deterministic simulation input variables and default values

Variable Name	Default Input Values	Units	Notes
runTimeLength	150	minutes	
truckStartPostions	[0, 0, 0, 8200, 8200, 8200, 8200]	feet	Distance from the start position 0
truckStartDirections	[0, 0, 0, 0, 0, 0, 0]		Position along grade from WasteDump, direction of truck 1 = unloaded(away from dump), -1 loaded(to dump),0 = delay
pulloutsSeg1	[-999]	feet	If the first value = -999 = free to pass in segment, if the first value = None = No passing in segment
pulloutsSeg2	[970]	feet	
pulloutsSeg3	[-999]	feet	
loadingTime	4	minutes	Assumed that only one truck can load at a time
dumpingTime	1.5	minutes	Assumed that multiple trucks can dump at a time
Lengthofsegments	[5000, 2000, 1200]	feet	
loadedSpeedsBySegment	[9, 6.5, 9]	miles per hour	
unloadedSpeedsBySegment	[9, 15, 9]	miles per hour	

The simulation was run for three situations based on the case study pit designs to understand how the trucks would interact in these circumstances. The outputs from simulating these situations are shown in Figure 6. The graph in the upper left of this figure shows an example simulation output; the rest of the graphs in this figure highlight Truck 1 to show the path of a single truck in the haulage cycle and expose where delays are being incurred within the haulage cycle. For each simulation, the haulage cycle was split into three segments, with Segment 1 including the haulage road going from the waste dump to the top of the in-pit ramp, Segment 2 running from the top of the ramp to where the toe of the ramp would be on average and Segment 3 running from the toe of the ramp to the center of the bench. In this simulation, position 0 ft. is the location of the dump, and location approximately 8,000 ft. is the center of the pit where the trucks are loaded. For each of the simulations in Figure 6, half of the trucks were started at the dump, and half of the trucks were started at the center of the pit to allow haulage cycles to reach a consistent cycle quickly.

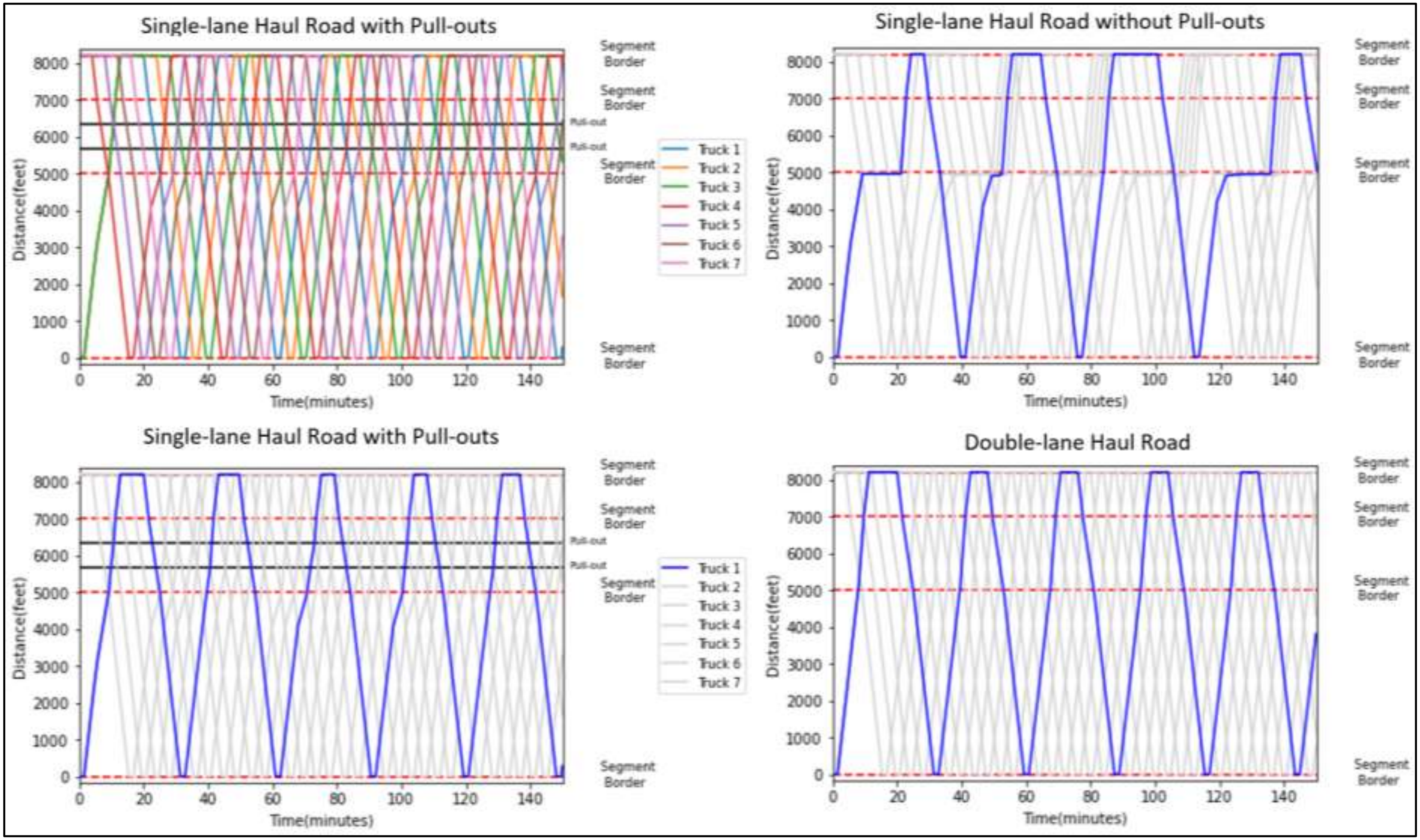

Figure 6: Case Study Deterministic Simulation Position by Time

Table 4: Production based on case study scenarios

Scenerio Descriptor	Loads Hauled in 12 hours	Percentage Production Decrease from Double-Lane Haulage
Double-lane	177	0%
2 pull-out	170	4%
1 pull-out	163	8%
Single-lane	125	29%

To better understand the sensitivity to the number of trucks and pull-outs, the simulation for the case study was run multiple times while recording the total loads hauled in 12 hours. As shown in the top graph in Figure 7, the inclusion of one pull-out allows for production similar to implementing two-lane haulage on the in-pit ramp. Table 4 provides production numbers for the case study simulations that ran in Figure 6. As pull-outs are added to a single-lane ramp, delays caused by adjusting speed to avoid collisions are reduced until the only major constraining factors negatively affecting production are loading and dumping delays. In Figure 7, it can be seen that as trucks and pull-outs are added to a haulage cycle, production increases until it reaches a maximum, which in this situation is approximately 175 loads hauled in 12 hours. This maximum appears to be caused by the loading delay, and since this simulation was run with the assumption of the use of a single loading unit, delays from loading could not be reduced without additional loading units. The locations of these different delays throughout the haulage cycle in multiple scenarios can be seen clearly in Figure 6.

Multiple pull-out sensitivity scenarios were run with progressively increasing distances to understand how an in-pit ramp of a larger distance would affect the number of pull-outs required to achieve production similar to double-lane haulage. The distances used for these scenarios were 2000 ft, 5000 ft, and 10000 ft. As expected, the number of pull-outs required to achieve similar production to double-lane haul roads is positively correlated to increased distances of the in-pit ramp.

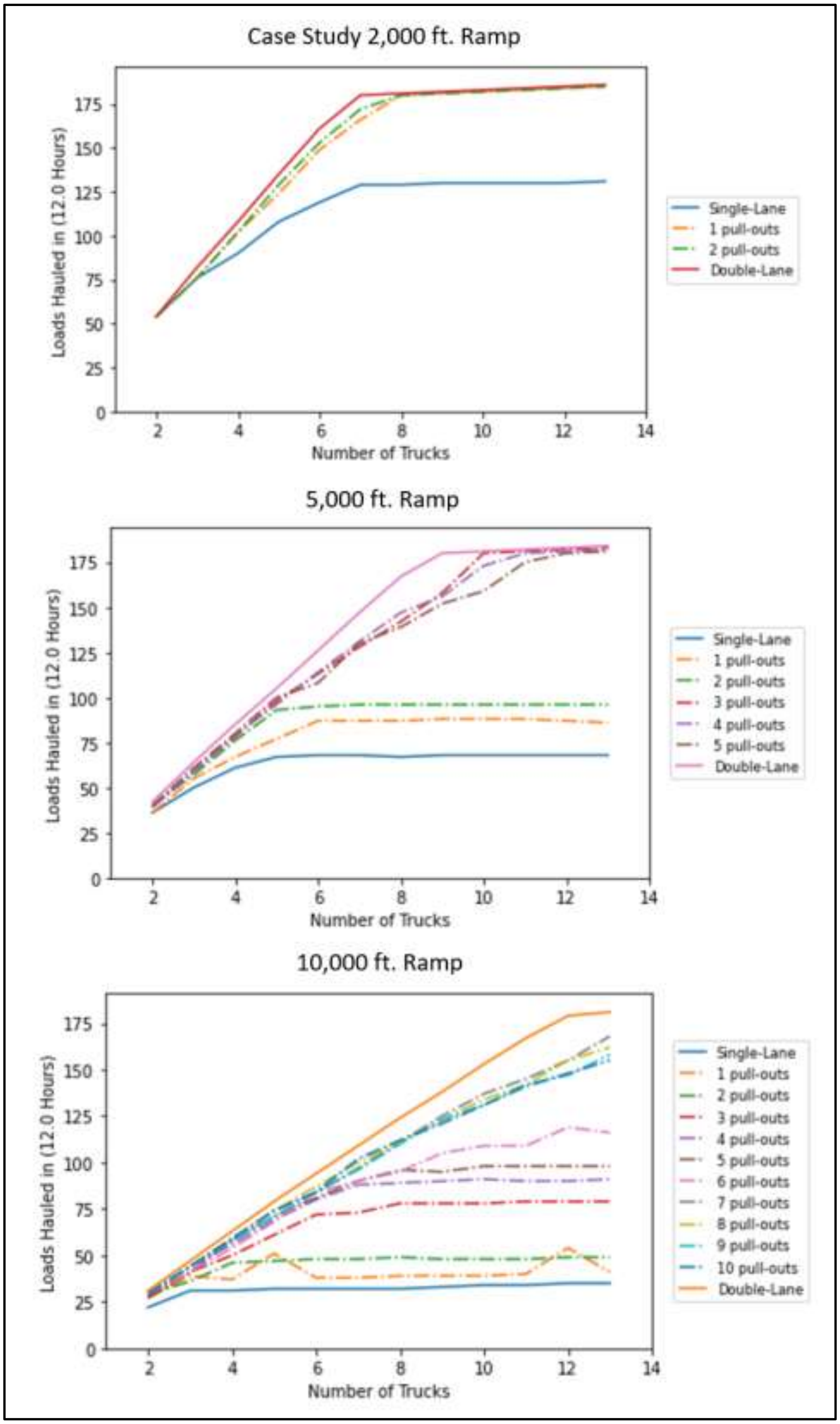

Figure 7: Deterministic Simulation Pull-out Sensitivity

In evaluating the simulation results, an error in the underlying code was discovered, however it has not been addressed in the attached code since it did not materially affect the results of the simulation. This error occurs when there is a long distance of no passing before the loading side of the haulage segment as shown in Figure 8. For seven trucks operating on segment lengths of 5000 ft., 10000 ft., and 1200 ft. with no passing on Segment 2, the error occurs approximately 14 times in 12 hours. In this scenario and the first interaction in Figure 8, the calculated speed of Truck 2 is set to the maximum speed after Truck 1 reaches the end of Segment 2. It is not until Truck 1 finishes loading until Truck 2 calculates optimal speed. By the time Truck 2 considers Truck 1, a collision is imminent. This error is only present when there is a large area where trucks cannot pass before the end of the segment. Also, this error explains why the 1 pull-outs and 2 pull-outs line cross in the bottom pull-out sensitivity graph for the 10,000 ft. ramp in Figure 7.

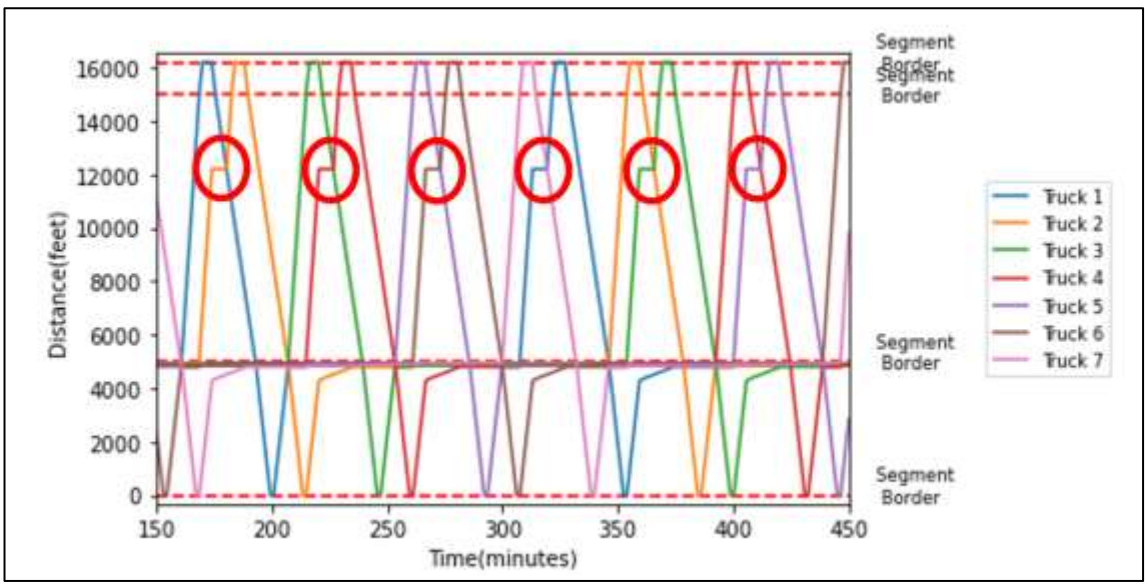

Figure 8: Simulation Error Example

Discussion

As shown in the methodology pit design section, reducing the road to a single-lane width results in a 17% decrease in the strip ratio, similar to what was found by Alegre et al. (2019). Both results show the strip ratio of the pit is significantly reduced when the road width is reduced. The Case study deterministic haulage simulations showed that by using two pull-outs, a single-lane road could achieve similar production, a 4% decrease, to a double-lane in-pit ramp. The research into single-track train scheduling provides insight into how automation could help unlock these benefits brought by reducing haul road widths.

By its very nature, automation means a fundamental change in how the overall mining process will operate (Parreira, 2013). Even though autonomous vehicles operate differently from manually operated vehicles, traditional parameters, rules of thumb, and laws continue to be used in modern-day mine designs without being questioned. Mine designs will need to be altered and optimized to play to the strengths of autonomous haulage and minimize the effects of any downsides introduced by this technology. One significant benefit of autonomous haulage is increased accuracy and consistency. This benefit can be exploited, and designs for these machines need less room for error. Reducing this room for error in haul roads can reduce the overall width of the haul road, which has a significant effect on the economics of the pit.

Pull-outs could be added to the design to operate two-way traffic on a single-lane haul road with efficient vehicle interactions. When scheduling haul trucks, priorities are assigned to each truck to help determine which trucks yield to other trucks to maximize the system's productivity. Generally, loaded haul trucks are going uphill, and the empty trucks are traveling downhill. In mining, the priority is given to the loaded vehicle, so the unloaded vehicle will have to adjust its speed to reach the pull-out before or preferably at the same time as an approaching loaded haul truck to minimize total delays incurred. It is of utmost importance that fully loaded trucks heading up and out of the mine never stop since they might stall and be unable to restart if stopped. Resolving such a situation can take time which would negatively affect costs, production time, and resources (Andersson, 2019). Furthermore, coming to a complete stop is not optimal as stopping increases the wear and tear on the trucks from using brakes and accelerating back to operating speed.

Using the simulation, the number and placement of pull-outs on a haulage ramp can be determined if the mine's desired productivity and number of trucks, along with the simulation inputs, are known. The pull-outs should be approximately equally spaced; however, these passing areas can be adjusted to places where the pit design requires widening for geological reasons. Although the pull-outs need to supply ample room for these trucks to pass, the additional width provided by a pull-out does not need to equal the full width of the truck, especially if the road is already two times the width of the haul trucks. Also, the pull-out length needs to account for both the speed of traffic and the size of the equipment operated on the ramp. If these areas are designed too short, and the trucks would have to stop. On the other hand, excess ground is excavated if these pull-outs are designed too long or too wide.

An artificial intelligence system or complex algorithm would need to be added to the current navigation systems to avoid collisions and adjust speeds for trucks to pass at pull-outs. The most optimal interaction would allow the higher priority truck to be unaffected by the interaction, and the lower priority truck would have minimal delay. A successful interaction would require the artificial intelligence or complex algorithm system knowing the locations, direction, and speed of all the trucks, at all times, along with the locations of the available passing areas. This system could be added to the current navigation system that determines what the truck does and how it interacts with its environment, regardless of whether it is on board the truck or centrally located.

Due to the precision and accuracy of the autonomous systems, cluster formation and shock wave propagation seen with human operation (i.e., the grouping of trucks at various points in the cycle) should be reduced. This distribution is optimal since most interactions will only involve two vehicles, and as mentioned in a paper by Anderson, the complications that result from multivehicle interactions can lead to collisions or inefficiencies. For example, when multiple vehicles are traveling together; the first vehicle will occupy the pull-out at the right time, but the second vehicle is left without any pull-out to use. This situation would result in a collision or one vehicle having to stop or reverse to the nearest available pull-out. Therefore, the autonomous system must consider the vehicles around them and determine how to avoid collisions and situations like this (Andersson, 2019).

Interactions between manually operated equipment, such as support equipment and light vehicles and autonomous vehicles, need to be considered when creating a haulage design. As mentioned by Anderson (2019), "Small personnel transports might have entirely different driving patterns to heavy trucks or machinery, and as such might present entirely different problems that need to be dealt with." Mines that run fully autonomous haul fleets have different protocols to ensure the safety of operators while ensuring consistent production, some even address this by creating additional roads for light vehicles. Voronov (2019) documents that mines operating fully autonomous haulage (as of 2019) will divide the mine into separate working areas for manned and unmanned vehicles. All vehicles entering the unmanned zone are equipped with satellite transceivers that provide tracking and communication with the autonomous system.

Two-way traffic on a decreased road width may also require increased road maintenance to ensure smooth roads and adequate dust control. A road at the width of two times the largest piece of equipment would allow for support equipment to maintain in-pit haul roads while reducing inefficient interactions with haulage vehicles. Along with support equipment, the potential for equipment to fail also needs to be considered, especially since autonomous haulage vehicles will stop in place if there are any issues with the vehicle. For example, if the communications network is not working correctly and an unmanned truck loses contact, the truck ceases to operate, stopping in place. In addition to network problems, the obstacle detection systems of unmanned trucks sometimes operate falsely, mistaking road bumps or small pieces of rock for obstacles, resulting in truck stoppage. In both of these situations, a manual visual inspection is required for the truck to resume operation. (Voronov, Voronov, & Makhambayev, 2020). This stoppage could be problematic when operating where there is no space to navigate around the equipment. Consideration of failed equipment needs to be accounted for in the design to allow for optimal operation. If this space is not accounted for, risk for increased delays is introduced since manually restarting the truck requires extra time and personnel oversight.

Connection issues are of great importance for large automation projects since autonomous vehicles rely on wireless communication and GPS. Even internet latency issues can have a significant impact on the operation of the mine. When considering implementing autonomous haulage for its many benefits, communication complexity problems are vitally important and need to be considered.

Conclusion

As the case study in this paper shows, road width can significantly affect strip ratio in an open pit mine; the wider the road is, the more flattening occurs to the highwall. In a deposit that already has a high strip ratio, this extra material mined will likely increase the strip ratio further, ultimately increasing the mining cost. The benefit of narrower haul roads could be unlocked with the use of autonomous haulage while keeping similar production rates to traditional two-lane roads. The advantages of this technology promote safer and more efficient hauling.

There are many considerations when contemplating operating autonomous haulage, especially when changing design fundamentals. When implementing autonomous haulage, the communication network needs to be carefully considered. If the communication network is lacking at a mine, then an attempt to operate autonomous haulage may fail. Along with an adequate network, understanding how autonomous haulage systems operate is necessary before implementing or specializing designs. To use unmanned haul trucks to their highest potential, one must take advantage of narrower haul roads. This paper only brings up some considerations that need to be made when contemplating operating autonomous haulage on a specialized design, such as reducing road width with pull-outs. Current design fundamentals will continue to be questioned with the advent of new technologies.

Future Work

Although this case study does not attempt to prove that haul road widths should be reduced simply due to automation, it does present an opportunity for future research and analysis. A survey of professionals with experience and knowledge of autonomous haulage in the mining industry should be conducted to gather helpful information about the intricacies of automated haulage. Ideally, these survey results would result in current information on autonomous vehicles' capabilities from multiple manufacturers.

A detailed haulage simulation could show how production would be affected by adapting open pit mine haulage road widths for autonomous operations. The deterministic simulation in this paper attempts to expose how the trucks would interact operating on different road designs. For a more detailed simulation, data from manual and autonomous haulage fleets could be used to run a more extensive haulage simulation.

In Deterministic Simulation, the error presented earlier should be fixed to provide a more accurate simulation of a mine's haulage cycle. In the code, it is possible for loaded trucks to calculate and consider possible collisions with trucks already on the ramp. If a collision is predicted, the truck will stop at the bottom of the ramp to avoid these collisions. Currently, the issue with this solution is stopping the unloaded trucks at the top of the ramp from traveling down the ramp while loaded trucks wait until all loaded trucks have traveled from the dump to the bottom of the ramp. In the future, a solution to this issue should be found and implemented in this simulation.

Other future research should focus on developing an algorithm or artificial intelligence system that controls the speed of an unmanned to ensure it reaches the pull-out at the correct time to maximize productivity. This paper has mentioned some, but not all of the many considerations that should be considered when creating this system to ensure the truck can operate on a reduced haul road width.

In the future, designs will be altered to leverage other technological advancements in the mining industry. Advances in haulage technology such as Komatsu's cab-less haulage truck concept could also affect mine designs to leverage its benefits. Voronov et al. (2020) describes this truck as an autonomous mining truck concept with a payload of 250 tons without a driver's cab. The machine has the ability to move forward and reverse at the same speed. Since all the wheels can turn independently, the turning radius is reduced. This reduction in turning radius will significantly decrease the time for placing for loading and dumping and the area required for loading and dumping zones.

References

Åhlén, M. (2014). A framework for traffic management and collision avoidance in underground mines.

Alegre, D., Peroni, R., & Aquino, E. (2019). The impact of haulroad geometric parameters on open pit mine strip ratio. *REM-International Engineering Journal, 72(1), 25-31.*

Andersson, C. (2019). Predictive Model for Traffic Control in Underground Mines.

Brundrett, S. (2014). INDUSTRY ANALYSIS OF AUTONOMOUS MINE HAUL TRUCK. SIMON FRASER UNIVERSITY.

Collins, J. L., Fytas, K., & Singhal , R. K. (1987). Design, construction and maintenance of surface mine haulage roads. *International Journal of Surface Mining, Reclamation and Environment*, 1:2, 137-145.

e-CFR. (2021, February 11). Retrieved from Electronic Code of Federal Regulations: https://www.ecfr.gov/cgi-bin/retrieveECFR?gp=&SID=7434e67c01d3aea69d5dd560ec3d6491&mc=true&n=pt30.1.56&r=PART&ty=HTML#sg30.1.56.h.sg22

Masabattula, V. (2011, February 11). *Haul road design*. Retrieved from slideshare: https://www.slideshare.net/venkoos/haul-road-design

Michaud, D. (2018, February 25). *Open Pit Mining*. Retrieved from 911Metallurgist: https://www.citationmachine.net/apa6/cite-a-website/new

Parreira, J. (2013). *An interactive simulation model to compare an autonomous haulage truck system with a manually-operated system (T).University of British Columbia.* Retrieved from https://open.library.ubc.ca/collections/ubctheses/24/items/1.0074111

Sakantsev, G. G., Cheskidov, V. I., Zyryanov, I. V., & Akishev, A. N. (2018). Validation of Slopes of Access Roads in Deep Open Pit Mining. *Journal of Mining Science*, 54(1), 77-84.

Voronov, Y., Voronov, A., & Makhambayev, D. (2020). Current State and Development Prospects of Autonomous Haulage at Surface Mines. *In E3S Web of Conferences (Vol. 174, p. 01028).* EDP Sciences.

Xu, X., Li, K., Yang, L., & Gao, Z. (2019). An efficient train scheduling algorithm on a single-track railway system. *Journal of Scheduling*, 22(1), 85-105.

Zhao, Z., & Bi, L. (2020). A New Challenge: Path Planning for Autonomous Truck of Open-Pit Mines in The Last Transport Section. *Applied Sciences*, 10(18), 6622.

Appendix

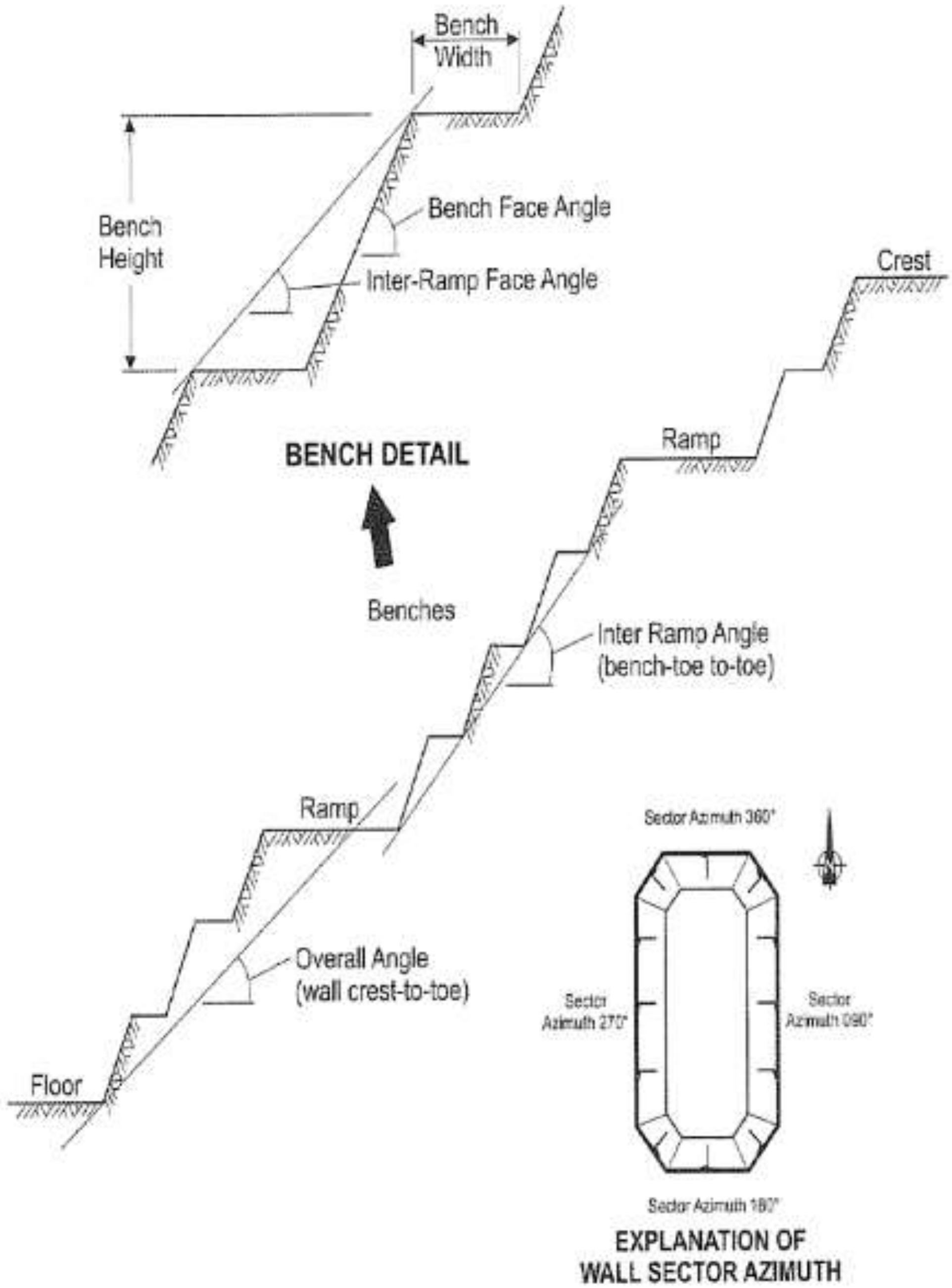

Figure 9: Open Pit Highwall Terminology (Michaud, 2018)

§56.9300 Berms or guardrails.

(a) Berms or guardrails shall be provided and maintained on the banks of roadways where a drop-off exists of sufficient grade or depth to cause a vehicle to overturn or endanger persons in equipment.
(b) Berms or guardrails shall be at least mid-axle height of the largest self-propelled mobile equipment, which usually travels the roadway.
(c) Berms may have openings to the extent necessary for roadway drainage.
(d) Where elevated roadways are infrequently traveled and used only by service or maintenance vehicles, berms or guardrails are not required when all of the following are met:
(1) Locked gates are installed at the entrance points to the roadway.
(2) Signs are posted warning that the roadway is not bermed.
(3) Delineators are installed along the perimeter of the elevated roadway so that, for both directions of travel, the reflective surfaces of at least three delineators along each elevated shoulder are always visible to the driver and spaced at intervals sufficient to indicate the edges and attitude of the roadway.
(4) A maximum speed limit is posted and observed for the elevated unbermed portions of the roadway. Factors to consider when establishing the maximum speed limit shall include the width, slope, and alignment of the road, the type of equipment using the road, the road material, and any hazardous conditions which may exist.
(5) Road surface traction is not impaired by weather conditions, such as sleet and snow, unless corrective measures are taken to improve traction.
(e) This standard is not applicable to rail beds.
[53 FR 32520, August 25, 1988, as amended at 55 FR 37218, September 7, 1990]

§56.9313 Roadway maintenance.

Water, debris, or spilled material on roadways which creates hazards to the operation of mobile equipment shall be removed.
(e-CFR, 2021)

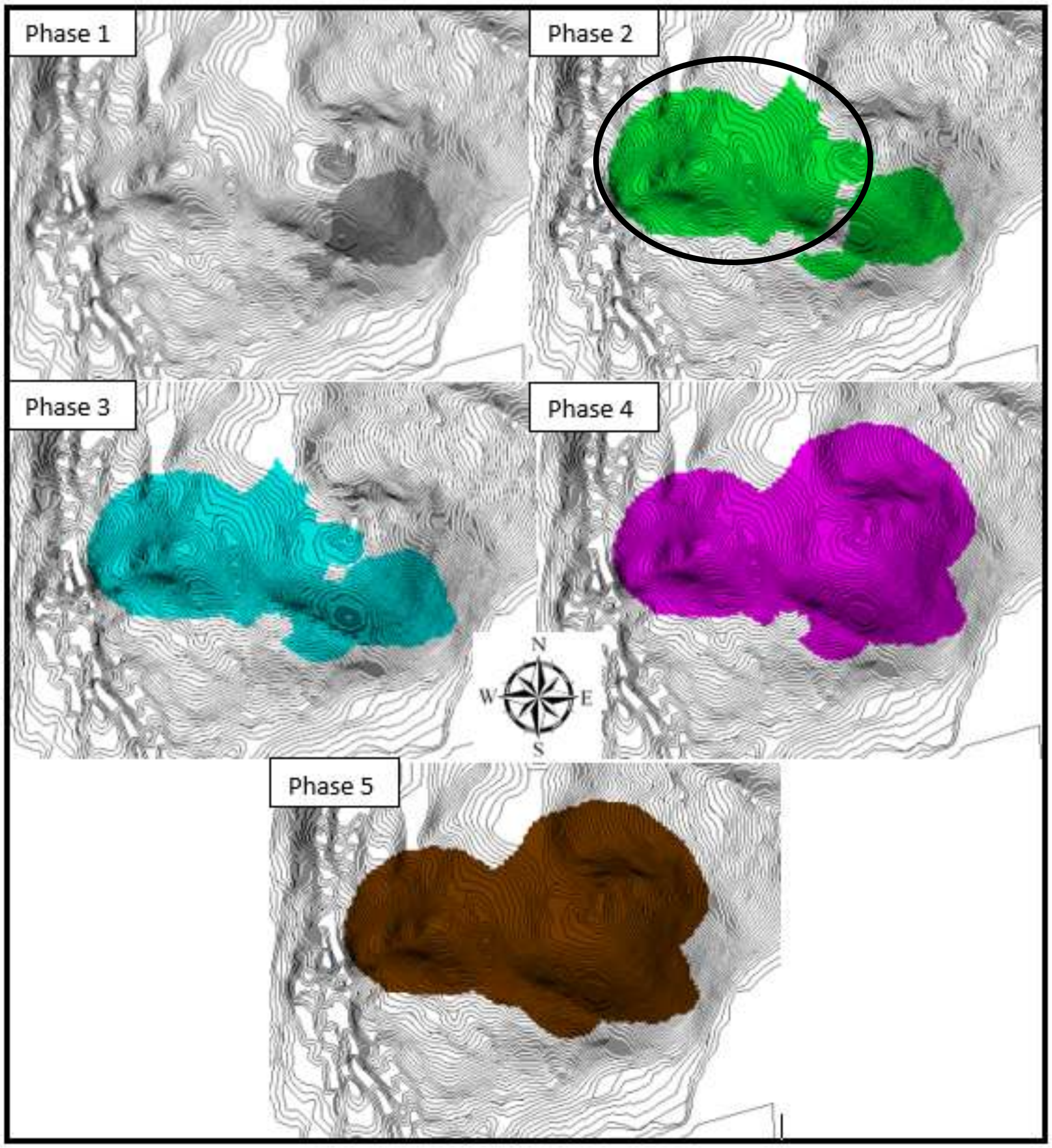

Figure 10: Case Study Deposit Pit Limit Analysis

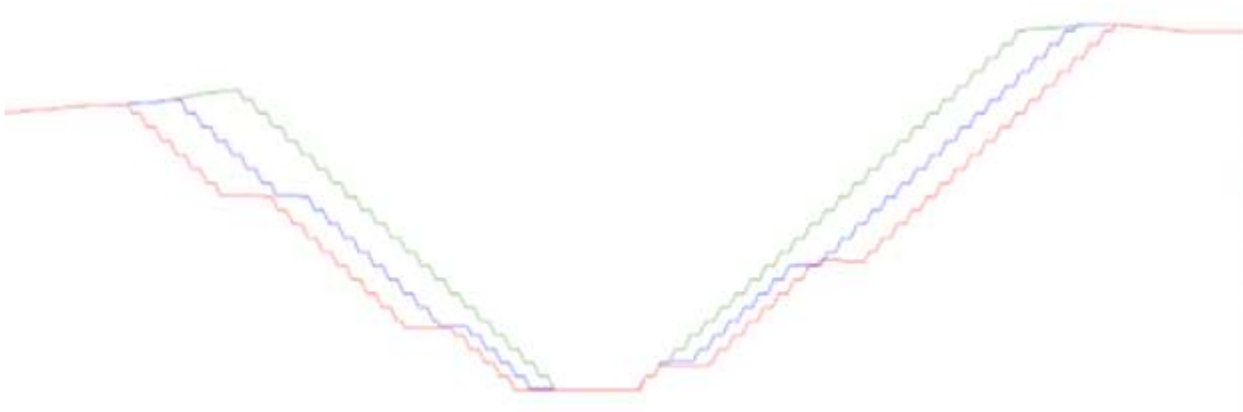

Figure 11: Pit Cross-section (180-0)

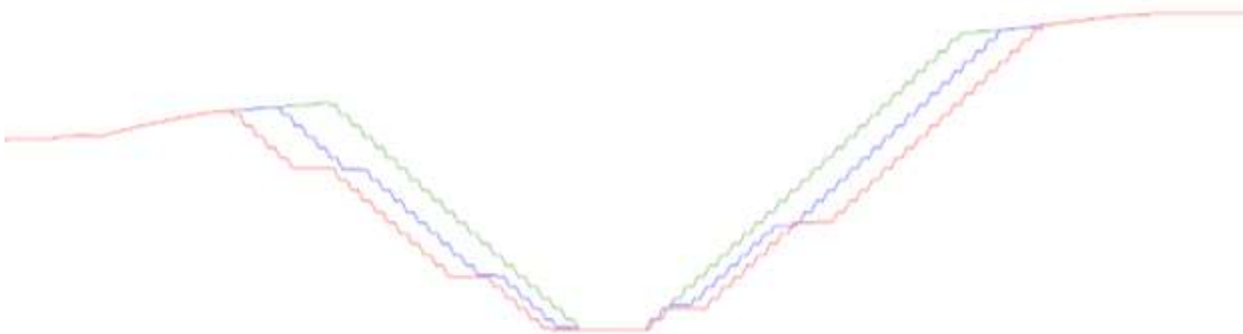

Figure 12: Pit Cross-section (202.5-22.5)

Figure 13: Pit Cross-section (225-45)

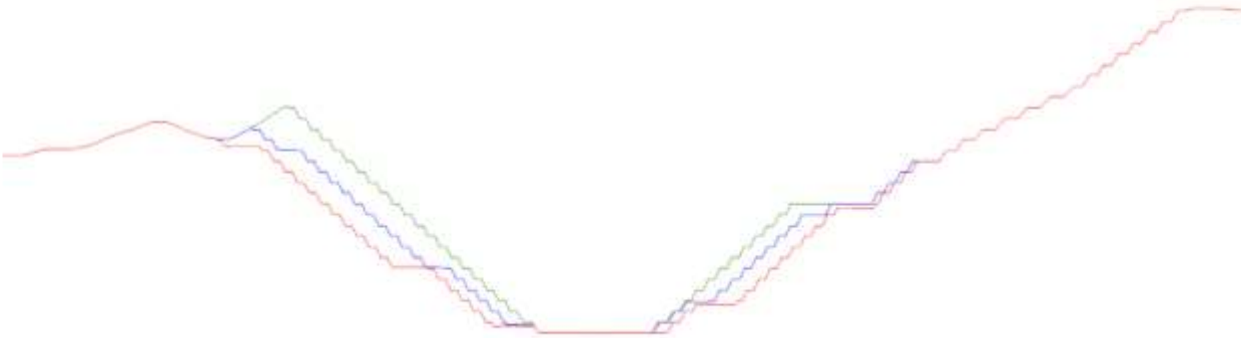

Figure 14: Pit Cross-section (247.5-67.5)

Figure 15: Pit Cross-section (270-90)

Figure 16: Pit Cross-section (292.5-112.5)

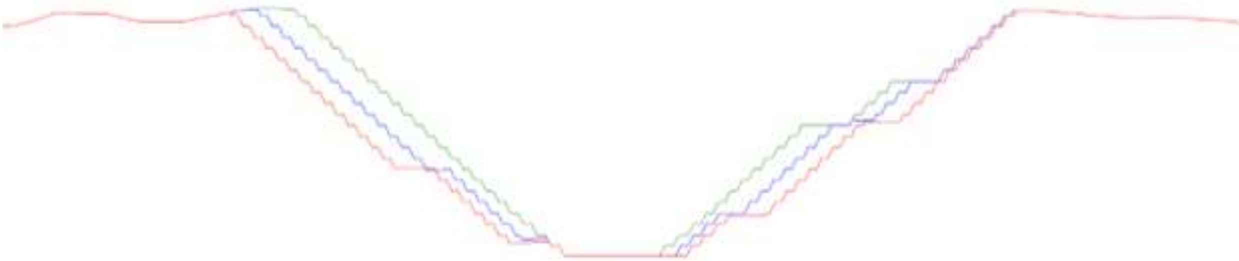

Figure 17: Pit Cross-section (315-135)

Figure 18: Pit Cross-section (337.5-175.5)

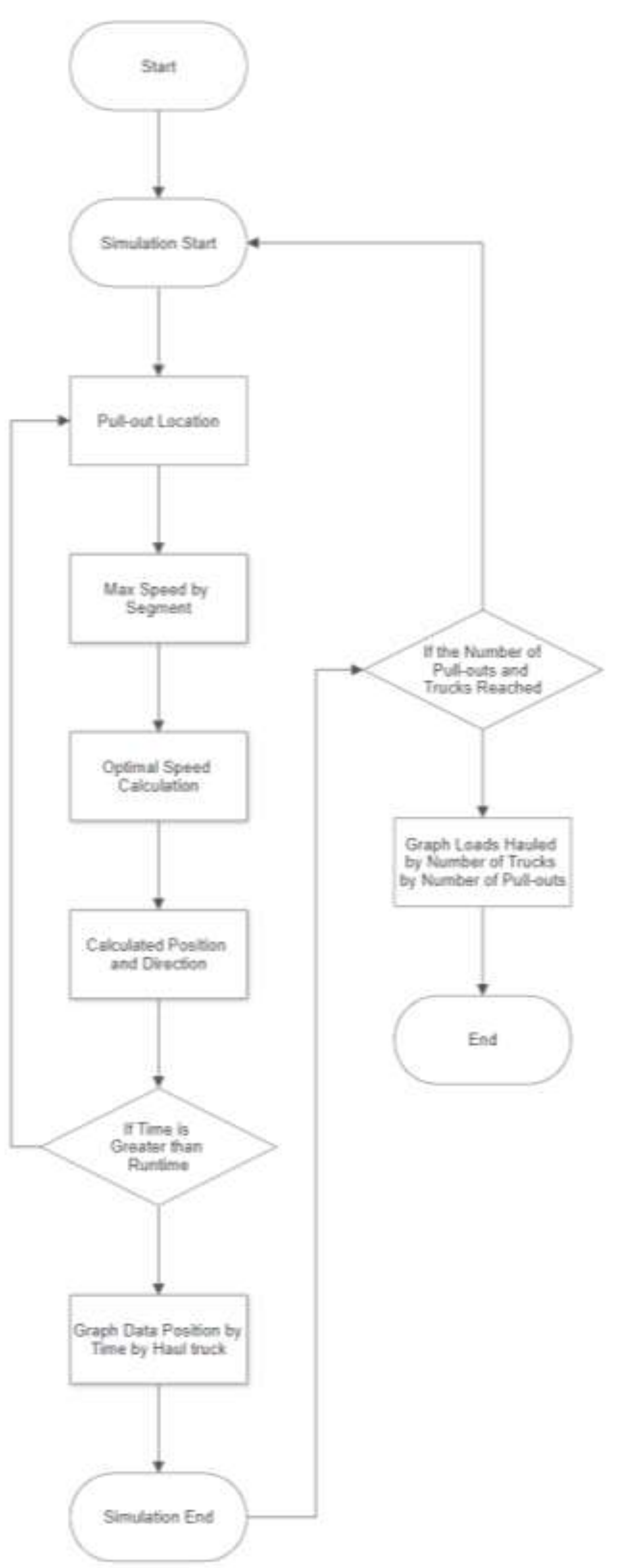

Figure 19: Flow Chart for Python Simulation Code

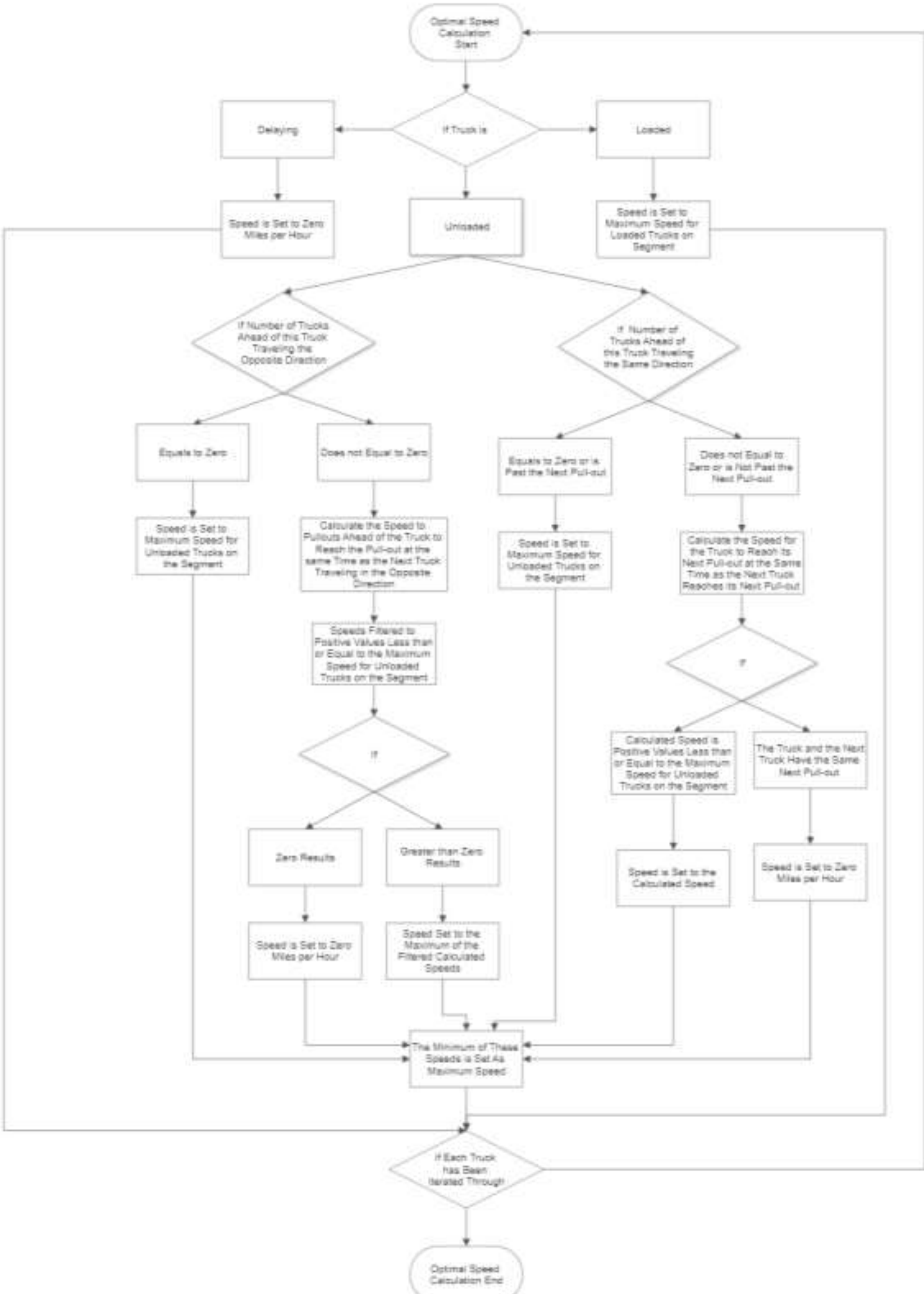

Figure 20: Flow Chart for Optimal Speed Calculation Code in the Simulation

Deterministic Simulation Python 3.8.8 Script

```
import numpy as np
import matplotlib.pyplot as plt
import csv

def Simulation(runTimeLength=150, # minutes positive float allowed
        truckStartPostions=[0, 0, 0, 8200, 8200, 8200, 8200], truckStartDirections=[0, 0, 0, 0, 0, 0, 0],# position
along grade from WasteDump, direction of truck 1 = unloaded(away from dump), -1 loaded(to dump),0 = delay
        pulloutsSeg1=[-999], pulloutsSeg2 =[970], pulloutsSeg3=[-999],# if first value = -999 = free to pass in
segment,  # if first value = None = No passing in segment,
        loadingTime=4, dumpingTime=1.5, # minutes positive float allowed
        lengthOfSegments=[5000, 2000, 1200], loadedSpeedsBySegment=[9, 6.5, 9],
unloadedSpeedsBySegment=[9, 15, 9]):

    trucks = np.array([np.array(truckStartPostions, dtype=float),
               np.array(truckStartDirections, dtype=float)])
    sgmntAttributes = np.array([np.array(lengthOfSegments, dtype=float),
                   np.array(loadedSpeedsBySegment, dtype=float),
                   np.array(unloadedSpeedsBySegment, dtype=float)])
    sgmntDist = np.cumsum(np.append([0], sgmntAttributes[0]))
    # Distance from the begining of the segment accepted values = 0 < x < segmentlength
    pulloutLoc = np.array([np.array(pulloutsSeg1),
                 np.array(pulloutsSeg2),
                 np.array(pulloutsSeg3)])
    # step size between calculations seconds
    step = 1 # seconds(Positive Whole Numbers)
    # time -> postion, direction, speed, delays -> truck
    simHist = np.zeros((1, 4, len(trucks[0])))
    simHist[0][[0, 1]] = trucks[:, trucks.argsort()][:, 0]

    def pulloutLocationCalc():
        pulloutDist = np.array([])
        truckLength = 50  # total length of a CAT 793 = 44 ft.
        for i, j in enumerate(pulloutLoc):
            if  j[0] != -999 and j[0] != None:
                pulloutDist = np.append(pulloutDist, sgmntDist[i] + np.cumsum(j))
            elif j[0] == -999:
                pulloutDist = np.append(pulloutDist, np.arange(sgmntDist[i], sgmntDist[i + 1], truckLength))
        pulloutDist = np.unique(np.append(pulloutDist, [sgmntDist[0], sgmntDist[-1]]))
        return pulloutDist

    def maxSpeedCalc():
        maxSpeed = np.array([], dtype=float)

        for truckNum in range(len(truckInfo[0])):
            for index, dist in enumerate(sgmntDist[1:]):
                if truckInfo[0][truckNum] <= dist:
                    if truckInfo[1][truckNum] == -1:
                        # moving in loaded
                        maxSpeed = np.append(maxSpeed, sgmntAttributes[[1, 2]][0][index])
                        break
                    elif truckInfo[1][truckNum] == 1:
```

```
                #moving in unloaded
                maxSpeed = np.append(maxSpeed, sgmntAttributes[[1, 2]][1][index])
                break
            else:
                maxSpeed = np.append(maxSpeed, 0)
                break
    return maxSpeed

def speedCalc():
    truckInfo[2] = maxSpeedCalc()
    pulloutLocations = pulloutLocationCalc()
    adjSpeed = np.array([], dtype=float)
    for truckNum in range(len(truckInfo[0])):
        if truckInfo[1][truckNum] == 1:
            tempArray = np.array([], dtype=float)
            # if there are no trucks ahead traveling in opposite direction set speed to max and skip to speed calc for
next truck
            # next closest truck location and speed opposite direction
            # trucksInFrontOppo = x1[:, x1[0] > truckInfo[0][truckNum]]
            trucksInFrontOppo = truckInfo[:, truckInfo[1, :] == -1][[0, 2]][:, truckInfo[:, truckInfo[1, :] == -1][[0, 2]][0] >
truckInfo[0][truckNum]]
            # next closest pullout locations
            pulloutNext = pulloutLocations[pulloutLocations > truckInfo[0][truckNum]]
            # next closest truck location and speed in the same direction
            trucksfInfrontWith = truckInfo[:, truckInfo[1] == 1][[0, 2]][:, truckInfo[:, truckInfo[1] == 1][[0, 2]][0] >
truckInfo[0][truckNum]]
            trucksfInfrontWithSort = trucksfInfrontWith[:, trucksfInfrontWith.argsort()][:, 0]
            if len(trucksInFrontOppo[0]) == 0:
                speed = truckInfo[2][truckNum]
            else:
                trucksInFrontOppoSort = trucksInFrontOppo[:,trucksInFrontOppo.argsort()][:, 0]
                # speed = [[dist truck1 to pullout] / [dist truck2 to pullout]] * [speed truck2]
                clacSpeedArray1 = ((pulloutNext - truckInfo[0][truckNum]) / (trucksInFrontOppoSort[0][0] -
pulloutNext)) * (trucksInFrontOppoSort[1][0])
                possibleClacSpeedArray1 = clacSpeedArray1[(clacSpeedArray1 <= truckInfo[2][truckNum]) &
(clacSpeedArray1 >= 0)]
                if len(possibleClacSpeedArray1) != 0:
                    speed = np.max(possibleClacSpeedArray1)
                else:
                    speed = 0
            tempArray = np.append(tempArray, speed)
            if len(trucksfInfrontWith[0]) == 0 or trucksfInfrontWithSort[0, 0] > pulloutNext[1]:
                speed = truckInfo[2][truckNum]
            else:
                NextTruckPulloutNext = pulloutLocations[pulloutLocations >= trucksfInfrontWithSort[0][0]]
                # speed = [[dist truck1 to its next closest pullout] / [dist truck2 to its next closest pullout]] * [speed
truck2]
                # speed for truck1 to reach the its next pullout at the same time as truck2 reaches its next pullout
                clacSpeed2 = ((pulloutNext[0] - truckInfo[0][truckNum]) / ( NextTruckPulloutNext[0] -
trucksfInfrontWithSort[0][0])) * (trucksfInfrontWithSort[1][0])
                # if if truck 1 and tuck 2 have the same next pullout then truck1 speed is set to 0
                if (pulloutNext[0] - truckInfo[0][truckNum]) >= (trucksfInfrontWithSort[0][0] - truckInfo[0][truckNum]):
                    # speed = (trucksfInfrontWith[1][0]) / 2
```

```
            speed = 0
          #if the calculated speed is less than the trucksmax speed and greater than or equal to 0
          elif (clacSpeed2 <= truckInfo[2][truckNum]) & (clacSpeed2 >= 0):
            speed = clacSpeed2
        tempArray = np.append(tempArray, speed)
        speed = np.min(tempArray)
      elif truckInfo[1][truckNum] == -1:
        speed = truckInfo[2][truckNum]
      else:
        speed = 0
      adjSpeed = np.append(adjSpeed, speed)
    return adjSpeed

  def positionAndDirection():
    truckInfo[2] = speedCalc()
    trucksLoaded = 0
    trucksDumped = 0
    mphToFps = ((1 / 60) * (1 / 60) * 5280)
    for truckNum in range(len(truckInfo[0])):
      if truckInfo[1][truckNum] == 1:
        positionTemp = truckInfo[0][truckNum] + (step * truckInfo[2][truckNum] * mphToFps)
        directionTemp = 1
        if positionTemp > sgmntDist[-1]:
          positionTemp = sgmntDist[-1]
          directionTemp = 0
          trucksLoaded += 1
      elif truckInfo[1][truckNum] == -1:
        positionTemp = truckInfo[0][truckNum] - (step * truckInfo[2][truckNum] * mphToFps)
        directionTemp = -1
        if positionTemp < sgmntDist[0]:
          positionTemp = sgmntDist[0]
          directionTemp = 0
          trucksDumped += 1
      else:
        # if postion is == end
        if truckInfo[0][truckNum] == sgmntDist[-1]:
          positionTemp = sgmntDist[-1]
          directionTemp = 0
          if truckInfo[3][truckNum] == np.max(truckInfo[:, truckInfo[0, :] == sgmntDist[-1]][3]):
            # number trucks that count delay is dependent on loading units which is assumed currently assumed
to be 1
            truckInfo[3][truckNum] = truckInfo[3][truckNum] + step
            # if the max time is greater than loading time
            if truckInfo[3][truckNum] > (abs(loadingTime) * 60):
              directionTemp = -1
              truckInfo[3][truckNum] = 0
        # if postion is == start
        elif truckInfo[0][truckNum] == sgmntDist[0]:
          truckInfo[3][truckNum] = truckInfo[3][truckNum] + step
          positionTemp = sgmntDist[0]
          # if the max time is greater than loading time
          if truckInfo[3][truckNum] > (abs(dumpingTime) * 60):
            directionTemp = 1
```

```
            truckInfo[3][truckNum] = 0
          else:
            directionTemp = 0
        # if the position is not end or start but direction is 0
        else:
          directionTemp = simHist[:, 1, truckNum][-1]
          # if there is not a previous direction that does not equal zero default direction 1 and drive to active face
          if directionTemp == 0:
            directionTemp = 1
          positionTemp = truckInfo[0][truckNum]
    truckInfo[0][truckNum] = positionTemp
    truckInfo[1][truckNum] = directionTemp
  return truckInfo, trucksLoaded, trucksDumped

def simulationPlot():
  plotRangeX = np.array([0, 150])
  # inputs
  t = np.array(range(0, len(simHist), step)) / 60
  s = np.transpose(simHist[:, 0])
  # lines and annotation
  for i, j in enumerate(s):
    plt.plot(t, j, label="Truck {}".format((i + 1)))
    # comment  ------^ and uncomment -----v to focus on one truck
    # if i == 0:
      # plt.plot(t, j, label="Truck {}".format((i + 1)), color="b", zorder=100)
    # else:
      # plt.plot(t, j, label="Truck {}".format((i + 1)), color="lightgray", zorder=5)
  for i, j in enumerate(pulloutLoc):
    if  j[0] != -999 and j[0] != None:
      x = sgmntDist[i] + np.cumsum(j)
      plt.hlines(x, 0, t[-1], colors="k")
      for a in x:
        plt.text(plotRangeX[1] + 2, a - 10, "Pull-out", color="k", fontsize="x-small")
  plt.hlines(sgmntDist, 0, t[-1], colors="r", linestyles="dashed")
  for i, j in enumerate(sgmntDist):
    plt.text(plotRangeX[1] + 12, sgmntDist[i] - 250, "Segment \n Border", fontsize="small")
  plt.xlabel("Time(minutes)")
  plt.ylabel("Distance(feet)")
  # plt.title("Case Study Haulage Simulation")
  plt.legend(loc="center right", bbox_to_anchor=(1.39, 0.5), fontsize="small")
  plt.axis([plotRangeX[0], plotRangeX[1], (sgmntDist[-1]) * -0.02, sgmntDist[-1] * 1.02])
  # output
  plt.show()

t = 0
trucksLoaded = 0
trucksDumped = 0
while t < (abs(runTimeLength) * 60):
  i = int((t / step))
  truckInfo = simHist[i].copy()
  x = positionAndDirection()
  simHist = np.append(simHist, np.array([x[0]]),axis = 0)
  trucksLoaded += x[1]
```

```
        trucksDumped += x[2]
        t += step
    simulationPlot()
    return trucksLoaded


# Simulation(pulloutsSeg2 =[None])
# Simulation()
# Simulation(pulloutsSeg2 =[667,667])
# Simulation(pulloutsSeg2 =[-999])

minTrucks = 2
maxTrucks = 13
maxPullouts = 10
timeToSimulate = 720 # 12 hours == 720 minutes
segementLengthsArray = [5000, 10000, 1200]

# current code only iterates through diffent number of pullouts for segment2 (in-pit ramp)
x1 = np.zeros(maxPullouts + 2)
for i in range(minTrucks, maxTrucks + 1):
    y = np.array([])
    for k, j in enumerate(range(maxPullouts + 2)):
        if j == 0:
            b = [None]
        elif k == maxPullouts + 1:
            b = [-999]
        else:
            b = [round(segementLengthsArray[1] / (j + 1), )] * j
        cycles = Simulation(runTimeLength=timeToSimulate,
                        truckStartPostions=(np.linspace(0,np.sum(segementLengthsArray),i)),
                        truckStartDirections=([0] * i),
                        lengthOfSegments=segementLengthsArray,
                        pulloutsSeg1=[-999], pulloutsSeg2=b, pulloutsSeg3=[-999])
        y = np.append(y, np.array([cycles]))
    x1 = np.vstack([x1, y])

def variableToCsv():
    # 2D list of variables (tabular data with rows and columns)
    inputVariable = x1
    # loadsHualedOutPut.csv gets created in the current working directory
    with open ('loadsHualedOutPut.csv', 'w', newline='') as csvfile:
        myWriter = csv.writer(csvfile, delimiter=',')
        myWriter.writerows(inputVariable)

variableToCsv()

# inputs
x = np.array(range(0, len(x1[-(len(x1) - 1):len(x1)]))) + minTrucks
y = np.transpose(x1[-(len(x1) - 1):len(x1)])
# lines and annotation
for i, j in enumerate(y):
    if i == 0:
        plt.plot(x, j, label="Single-Lane")
```

```
    elif i == len(y) - 1:
        plt.plot(x, j, label="Double-Lane")
    else:
        plt.plot(x, j, label="{} pull-outs".format((i)), linestyle="dashdot")
plt.xlabel("Number of Trucks")
plt.ylabel("Loads Hauled in ({} Hours)".format(timeToSimulate / 60))
# plt.title("Case Study Haulage Simulation")
plt.legend(loc="center right", bbox_to_anchor=(1.3, 0.5), fontsize="small")
plt.axis([minTrucks - 1, maxTrucks + 1, 0, np.max(y) + 10])
# output
plt.show()
```